Appl. No. 10/749,910
Amdt. dated October 5, 2007
Reply to Office Action of July 6, 2007

PATENT

REMARKS/ARGUMENTS

This Amendment is in response to the Office Action mailed July 6, 2007. Claims 1, 2, 5-8, 11-14, and 16 were pending in the present application. This Amendment does not add, cancel, or amend any claims, leaving pending in the application claims 1, 2, 5-8, and 11-14 and 16. Reconsideration of the rejected claims is respectfully requested.

I.   Rejection under 35 U.S.C. §112

Claims 1, 2, 5-8, 11-14, and 16 are rejected under 35 U.S.C. §112, first paragraph, as failing to comply with the written description requirement. In particular, the Office Action sets forth that it is not clear where certain limitations of the claims are found in the specification. Applicants respectfully submit that such limitations are presented in the specification, and will go through an example as follows:

On page 2, starting at line 24 of the summary, the specification states that "when a memory access request requires multiple data bursts on the memory bus" the data is stored from the multiple data bursts "in respective buffers", such that data retrieved from the buffers is "read from a first part of the first buffer, then from the other buffers, and finally from the remaining part of the first buffer." (*emphasis added*) Storing the data in the remaining part of the first buffer "avoids the need to occupy the memory bus with a new data burst" that would be stored in a respective buffer (page 2, lines 31-33). Thus, it can be seen that the beginning and end of the read request pull data from the same buffer.

For further detail, on page 6 starting at line 30, if the control logic determines that a wrapping burst is required, the control logic "calculates the number of SDRAM bursts required to fulfill the access request". The control logic then assigns one of the buffers in the SDRAM interface to each of the required bursts (page 7, lines 9-11). For a wrapping read request, the first SDRAM data burst may have data written thereto such that the first SDRAM data bursts contains "the data required at the end of the request, as well as the data required at the start" (page 9, lines 8-12). Referring to FIG. 4, where data is to be returned to the requesting device, data is first read out of "successive sub-buffers 601 in the first buffer 60, as shown by the solid arrow 70" (page 9,

lines 14-22). The controller keeps track of the AHB address, and is able to determine when the next item of data to be returned comes from the second burst and "must be retrieved from the second buffer 62" (p. 9, lines 17-22). The control logic then "records the value of a pointer 72, indicating the sub-buffer 602 from which data was next to be retrieved" (p. 9, lines 24-26). Data is then read out of the second buffer until the last buffer of that request. When all of the data is read from the last buffer, and the requesting device is still requesting more data (the address has wrapped), "the control logic returns to the sub-buffer 602" of the first buffer "indicated by the pointer 72" and reads out data from the first buffer until the end of the buffer to provide the "final data requested by the requesting device" (page 9, line 35-page 10, line 6). Thus, the data "required at the end of the data transfer to the requesting device was effectively cached in the [first] buffer 60 until is was required", which "avoids the need to transfer the data in a separate SDRAM burst, and therefore makes better use of the bandwidth of the memory bus 60" (page 10, lines 8-12).

The Office Action makes the assumption (set forth explicitly on page 4 of the Office Action) that the beginning and end of the request were stored in different sub-buffers of the plurality of buffers, and not in a single buffer (see OA pp. 3-4). This would require the same data burst to be requested at different times, providing inefficient use of bandwidth, as set forth in the background of the Applicants' specification on page 2, at lines 8-16). As set forth above, storing the required end data "in the remaining part of the first buffer" avoids the need to occupy the memory bus with a new data burst.

As such, Applicants respectfully submit that the limitations recited in the claims are correctly supported by the specification and meet the written description requirement. It is respectfully submitted, therefore, that the assumption that the beginning and the end of the request are stored in, and read from, separate buffers is not a correct interpretation and in fact improperly reads out limitations in the claims. It is further respectfully submitted that, using the proper interpretation as set forth above, none of the references teach or suggest such limitations and the claims should thus be allowable over these references.

Applicants therefore respectfully request that the §112 rejections with respect to these claims be withdrawn.

II.   Rejection under 35 U.S.C. §103

(a) *Gray, Becker,* and *Nguyen*

Claims 1, 2, 7-8, 13-14, and 16 are rejected under 35 U.S.C. 103(a) as being obvious over *Gray* (U.S. Pat. No. 6,816,923) in view of *Becker* (U.S. Pat. No. 6,950,884) and further in view of *Nguyen* (US 5,335,326). Applicants respectfully submit that these references do not teach or suggest each element of these claims.

As discussed above, the Office Action interprets the claims in a way that was not intended (and in fact reads limitations out of the claims) and is not as was set forth in the specification. In particular, the Office Action assumes that the beginning and end data for a wrapping request were stored in separate buffers, and not sub-buffers of a single, common buffer. While the rejections below might support rejection of the claims based on such an interpretation, the correct interpretation as set forth above utilizes sub-buffers of a <u>single buffer</u> that can be written at substantially the same time from the same burst, and thus do not require a separate burst. Such limitations are not rendered obvious by these references, and the claims as pending should be allowable over these references.

Although utilizing the language as set forth in the claims and supported by the specification should be sufficient to overcome the rejections, Applicants will address the rejections for purposes of completeness.

For example, Applicants' claim 1 recites a memory controller, comprising:

> at least one bus interface, each bus interface being for connection to at least one respective device for receiving memory access requests;
> a memory interface, for connection to a memory device over a memory bus;
> a plurality of buffers in the memory interface; and
> control logic, for placing received memory access requests into a queue of memory access requests,
> wherein, in response to a received memory access request requiring multiple data bursts over the memory bus, each of said multiple data bursts is assigned by the control logic to a respective buffer of the plurality buffers in the memory interface, and data from each of said multiple data bursts is stored by the memory interface in the respective buffer, and
> wherein, **for a wrapping memory access request requiring multiple buffers, data required for each of a beginning and an end of the wrapping memory access request are**

> **assigned to respective sub-buffers of a single respective buffer by the control logic**, the
> beginning and end data for the memory access request being stored in the respective sub-buffers
> by the memory interface, **the storing of the beginning and end data in a single buffer avoiding
> the need for an additional data burst to obtain the end data**, and
> > **wherein the control logic records a value of a pointer indicating a first sub-buffer of
> the single buffer storing the end data, such that the control logic is able to return to the
> indicated sub-buffer to retrieve the end data from the single buffer**

(emphasis added). Such limitations are neither taught nor suggested by these references,
individually or in combination.

For example, *Gray* teaches a direct memory access (DMA) system including a DMA
engine that includes a data reservoir having a number of memory buffers in order to consolidate
memory buffers for the various devices into the DMA reservoir, the reservoir including portions
that correspond to different devices (col. 2, lines 34-47; col. 3, line 64-col. 4, line 18). The use
of the consolidated memory reservoir provides the ability to centralize addressing and provide
each device with data in a timely manner and with increased bandwidth (col. 2, lines 34-56).
Gray consolidates memory into buffers in the data reservoir (shown in Fig. 3; also col. 2, lines
34-47; col. 4, lines 10-12).

As recognized in the Office Action, *Gray* does not teach or suggest a request requiring
multiple data bursts and data required for a beginning and an end of the request being stored in a
single buffer, for a wrapping memory access request requiring multiple buffers. Further, *Gray*
does not teach or suggest control logic assigning different portions of a single request from a
single device to different sub-buffers of the single buffer, and recording a pointer so that the
control logic can come back to the single buffer in order to obtain the end portion of the data
without requiring an additional data burst as recited in Applicants' claim 1 as amended. For at
least these reasons, *Gray* cannot render obvious Applicants' claim 1 or the claims that depend
therefrom.

Becker does not make up for the deficiencies in *Gray* with respect to Applicants' claim 1.
Becker teaches a DMA device for transferring data between two processors (col. 1, lines 54-60;
col. 4, lines 32-58). *Becker* teaches using "cyclic memories," wherein read and write access
between the two processors takes place in rising or falling memory block order (col. 8, line 65-
col. 9, line 6). For example, when the last block of one control information memory is reached,

the first block is automatically written to as the next memory block (col. 8, line 65-col. 9, line 6). *Becker* does not, however, teach or suggest control logic assigning different portions of a single request from a single device to different sub-buffers of the single buffer, and recording a pointer so that the control logic can come back to the single buffer in order to obtain the end portion of the data without requiring an additional data burst as recited in Applicants' claim 1 as amended. For at least these reasons, *Becker* cannot render obvious Applicants' claim 1 or the claims that depend therefrom, either alone or in combination with *Gray*.

Even if for sake of argument there were motivation to combine *Gray* and *Becker*, which Applicants do not believe as set forth previously, combining *Becker's* circular buffer with *Gray's* device-specific buffers would not result in a memory controller with control logic assigning different portions of a single request from a single device to different sub-buffers of the single buffer, and recording a pointer so that the control logic can come back to the single buffer in order to obtain the end portion of the data without requiring an additional data burst as recited in Applicants' claim 1 as amended.

Nguyen does not make up for the deficiencies in either reference with respect to these claims. *Nguyen* teaches a bus-to-bus interface including a central buffer means including first and second FIFO devices, utilizing a pointer means to traverse a circular queue in the FIFO devices slot by slot (col. 1, line 59-col. 2, line 49). *Nguyen* does not, however, teach or suggest control logic assigning different portions of a single request from a single device to different sub-buffers of a single buffer, and recording a pointer so that the control logic can come back to the appropriate location in the single buffer in order to obtain the end portion of the data without requiring an additional data burst as recited in Applicants' claim 1 as amended. As such, *Nguyen* cannot render obvious Applicants' claim 1 as amended, either alone or in combination with *Gray* and *Becker*.

There is no motivation in these references to combine the references. Even if for sake of argument the references were combined, the result would not reach the limitations or advantages of Applicants' claimed invention, as combining *Becker's* circular buffer with *Gray's* device-specific buffers and *Nguyen's* FIFO pointer would not result in a memory controller with control logic assigning different portions of a single request from a single device to different sub-buffers

of a single buffer, and recording a pointer so that the control logic can come back to the single
buffer in order to obtain the end portion of the data without requiring an additional data burst as
recited in Applicants' claim 1 as amended. The resulting device would still use a separate buffer
for the end portion, which would require an additional burst.

For at least these reasons, the combination of *Gray*, *Becker*, and *Nguyen* fails to teach or
suggest each element of Applicants' claim 1, such that these references cannot render obvious
Applicants' claim 1 or the claims that depend therefrom. Independent claims 7 and 13 recite
limitations that similarly are not rendered obvious by these references for reasons including those
cited above. Applicants therefore respectfully request that the rejection with respect to claims 1,
2, 7-8, 13-14, and 16 be withdrawn.

(b) *Gray, Becker, Nguyen, and Kuronuma*

Claims 5 and 11 are rejected under 35 U.S.C. 103(a) as being obvious over *Gray, Becker*,
and *Nguyen*, and further in view of *Kuronuma* (U.S. Pat. No. 6,859,848). Claims 5 and 11
depend from claims 1 and 7, respectively, which are not rendered obvious by *Gray, Becker*, and
Nguyen as discussed above. *Kuronuma* does not make up for the deficiencies in *Gray, Becker*,
and *Nguyen* with respect to these claims. *Kuronuma* teaches a memory control system for
sequentially accessing an arbitrary address in an SDRAM circuit (col. 4, lines 22-25), and is
cited as teaching sequential access to an SDRAM (OA page 10). *Kuronuma* does not, however,
teach or suggest a memory controller with control logic assigning different portions of a single
request from a single device to different sub-buffers of the single buffer, and recording a pointer
so that the control logic can come back to the single buffer in order to obtain the end portion of
the data without requiring an additional data burst. As such, *Kuronuma* cannot render obvious
claims 1 and 7, or dependent claims 5 and 11, alone or in any combination with *Gray, Becker*,
and *Nguyen*. Applicants therefore respectfully request that the rejection with respect to claims 5
and 11 be withdrawn.

(c) *Gray, Becker, Nguyen* and *Microsoft*

Claims 6 and 12 are rejected under 35 U.S.C. 103(a) as being obvious over *Gray, Becker*, and *Nguyen*, and further in view of *Microsoft* ("Microsoft Computer Dictionary", 2002 p. 469). Claims 6 and 12 depend from claims 1 and 7, respectively, which are not rendered obvious by *Gray, Becker*, and *Nguyen* as discussed above. *Microsoft* does not make up for the deficiencies in *Gray, Becker*, and *Nguyen* with respect to these claims. *Microsoft* is cited as teaching SDRAM as a common type of RAM (OA page 11). *Microsoft* does not, however, teach or suggest a memory controller with control logic assigning different portions of a single request from a single device to different sub-buffers of the single buffer, and recording a pointer so that the control logic can come back to the single buffer in order to obtain the end portion of the data without requiring an additional data burst. As such, *Microsoft* cannot render obvious claims 1 and 7, or dependent claims 6 and 12, alone or in any combination with *Gray, Becker*, and *Nguyen*. Applicants therefore respectfully request that the rejection with respect to claims 6 and 12 be withdrawn.

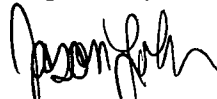## II.   Amendment to the Claims

Unless otherwise specified or addressed in the remarks section, amendments to the claims are made for purposes of clarity, and are not intended to alter the scope of the claims or limit any equivalents thereof. The amendments are supported by the specification and do not add new matter.

CONCLUSION

In view of the foregoing, Applicants believe all claims now pending in this Application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at 925-472-5000.

Respectfully submitted,

Jason D. Mohr
Reg. No. 48,163

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: 925-472-5000
Fax: 415-576-0300
JDL:slh
61167974 v1